KING SAUD UNIVERSITY
COLLEGE OF COMPUTER & INFORMATION SCIENCES
DEPT OF COMPUTER SCIENCE

CSC311 Algorithms Design and Analysis
Second Semester 1444 AH
Instructor: Dr. Mohamed Maher Ben Ismail

Tutorial #3

By. 3meer

Problem 1

For each algorithm listed below,

give a recurrence that describes its worst-case running time, and give its worst-case running time using θ-notation.

You need **not** justify your answers.

(a) Merge sort
Recurrence: $T(n) = 2T(n/2) + O(n)$
Worst-case : $\Theta(n \log n)$

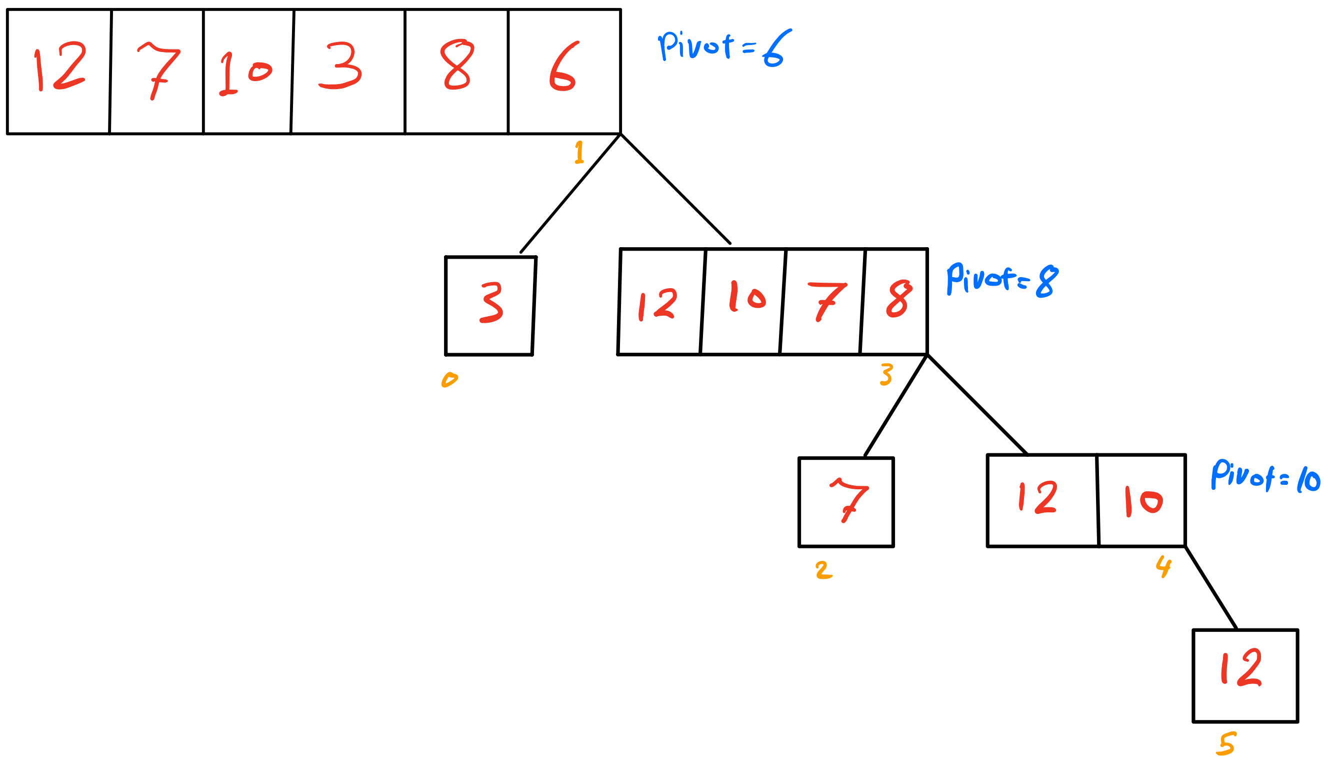
(b) Insertion sort
recurrence: $T(n) = T(n-1) + O(n)$
worst-case: $\Theta(n^2)$

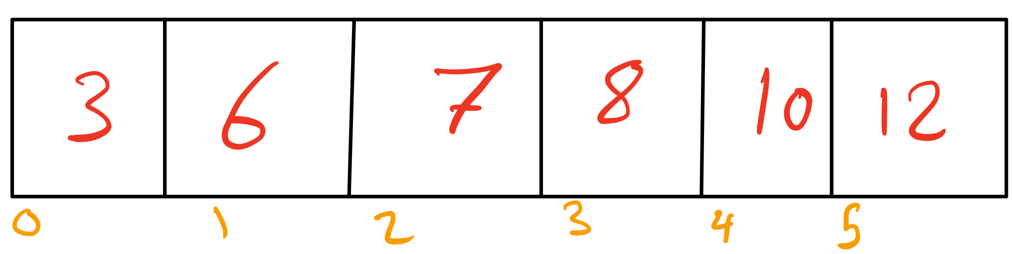
(c) Merge's algorithm
No Recursive.

(d) Binary Search
recurrence: $T(n) = T(n/2) + O(1)$
worst-case: $\Theta(\log n)$

Problem 2

Do Quicksort on the input numbers 12, 7, 10, 3, 8, 6. Illustrate the execution using a binary tree where each node is an array or subarray when calling the PARTITION procedure. The two children of a node are the subarrays immediately after partitioning their parent array.





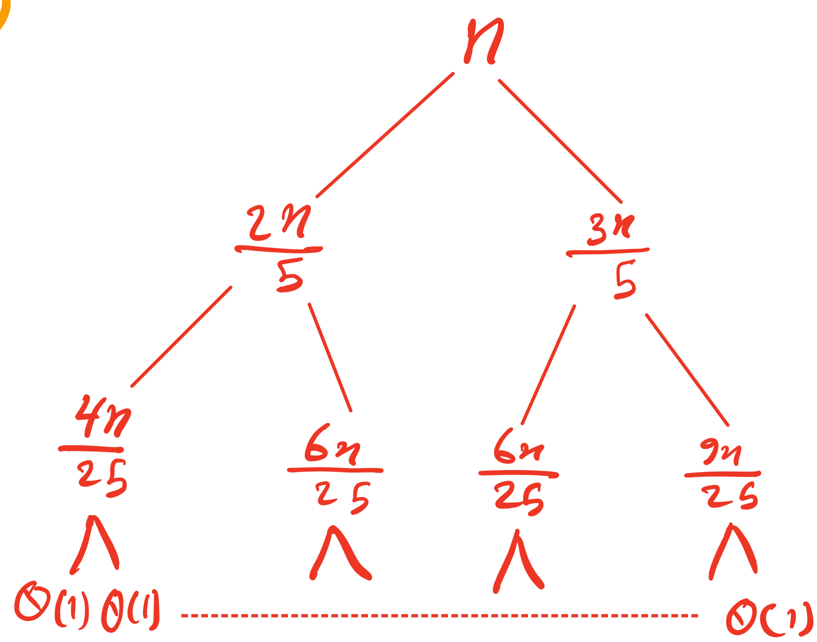
Problem 3

Consider a variation of MergeSort which divides the list of elements into two lists of size 2/5 and 3/5, recursively at each step, instead of dividing it into halves. The Merge procedure does not change.

(a) Give a recurrence relation for this algorithm
(b) Draw a recursion tree for the algorithm
(c) Using the recursion tree, explain how the worst case upper bound is O(nlogn).

a) $T(n) = T(2n/5) + T(3n/5) + n$

b)



c)

h: height

stop

$n = \left(\frac{5}{3}\right)^K$

$K = \log_{\frac{5}{3}} n$

$K = h$

$T(n) \leq n \cdot \log_{\frac{5}{3}} n$

$\leq c \cdot g(n)$

$\leq n \cdot \log_{\frac{5}{3}} n$

$C = 1 \qquad n_0 = \frac{5}{3} \qquad O(n \log_{\frac{5}{3}} n)$